ML model for optimized interface between dispatch and crushing process

Presenter: Bruno Galdos

March 2025

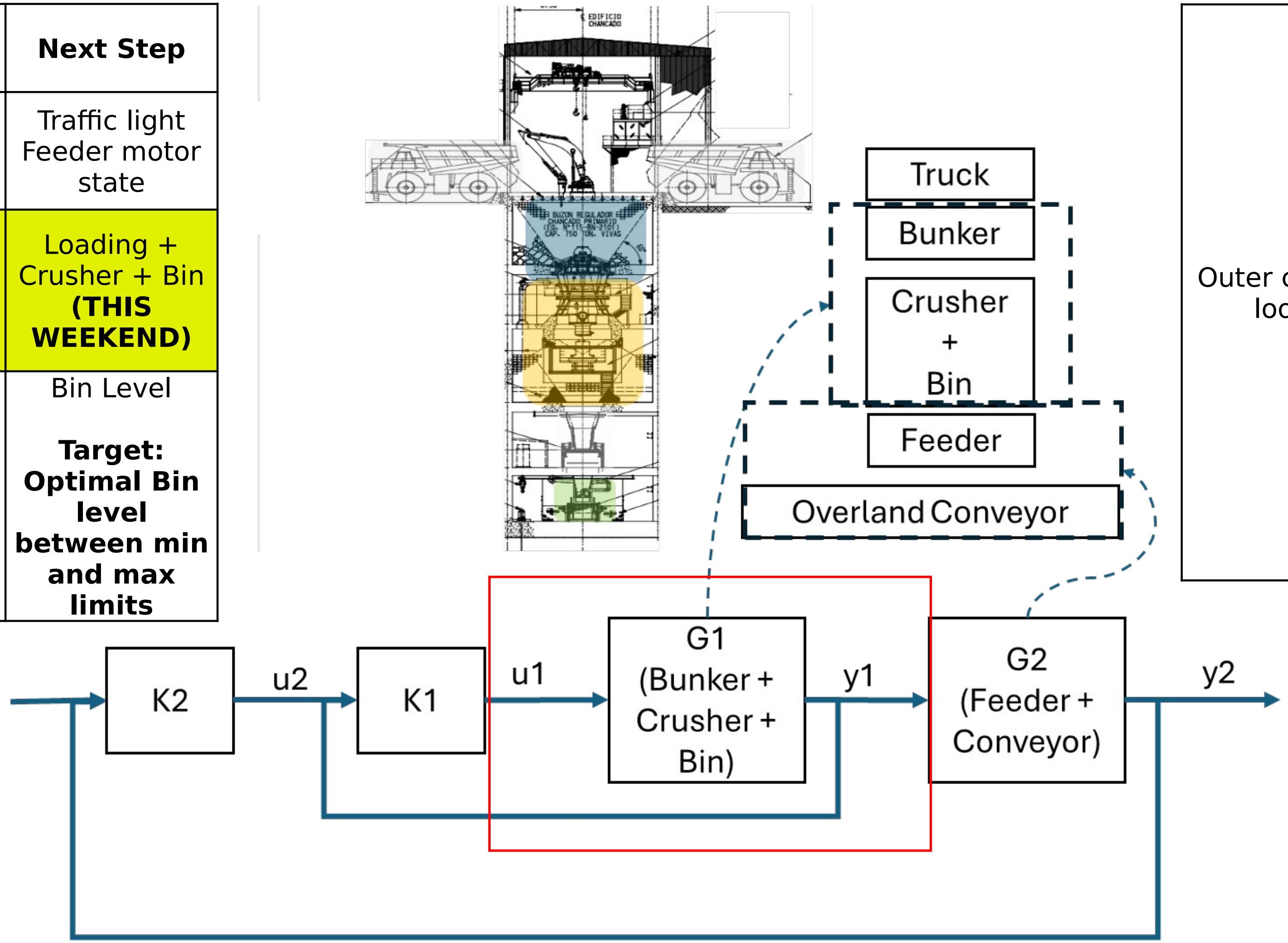
INNOMOTICS

Structure of the model

INNOMOTICS

Approach: Identify the dependencies of each component

Inner closed loop	K1 (Controller)	**Next Step**
	u1 (Input)	Traffic light Feeder motor state
	G1 (Model)	Loading + Crusher + Bin **(THIS WEEKEND)**
	y1 (Output)	Bin Level **Target: Optimal Bin level between min and max limits**

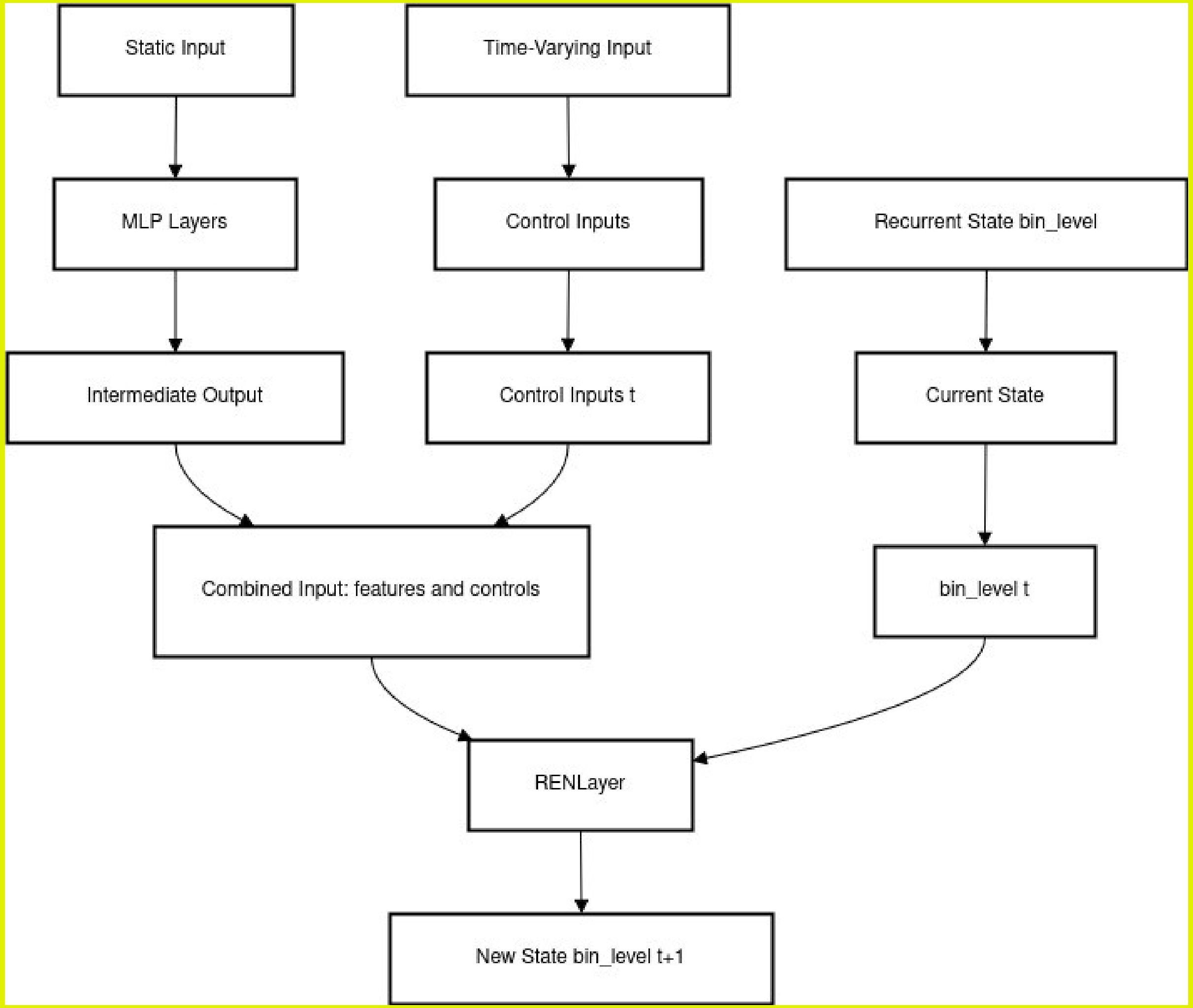


Outer closed loop	K2 (Controller)	**Next step**
	u2 (Input)	Traffic light Feeder motor state
	G2 (Model)	Feeder + Conveyor
	y2 (Output)	Output of conveyor **Target: Optimized Flow 10.500 tons/hr**

INNOMOTICS

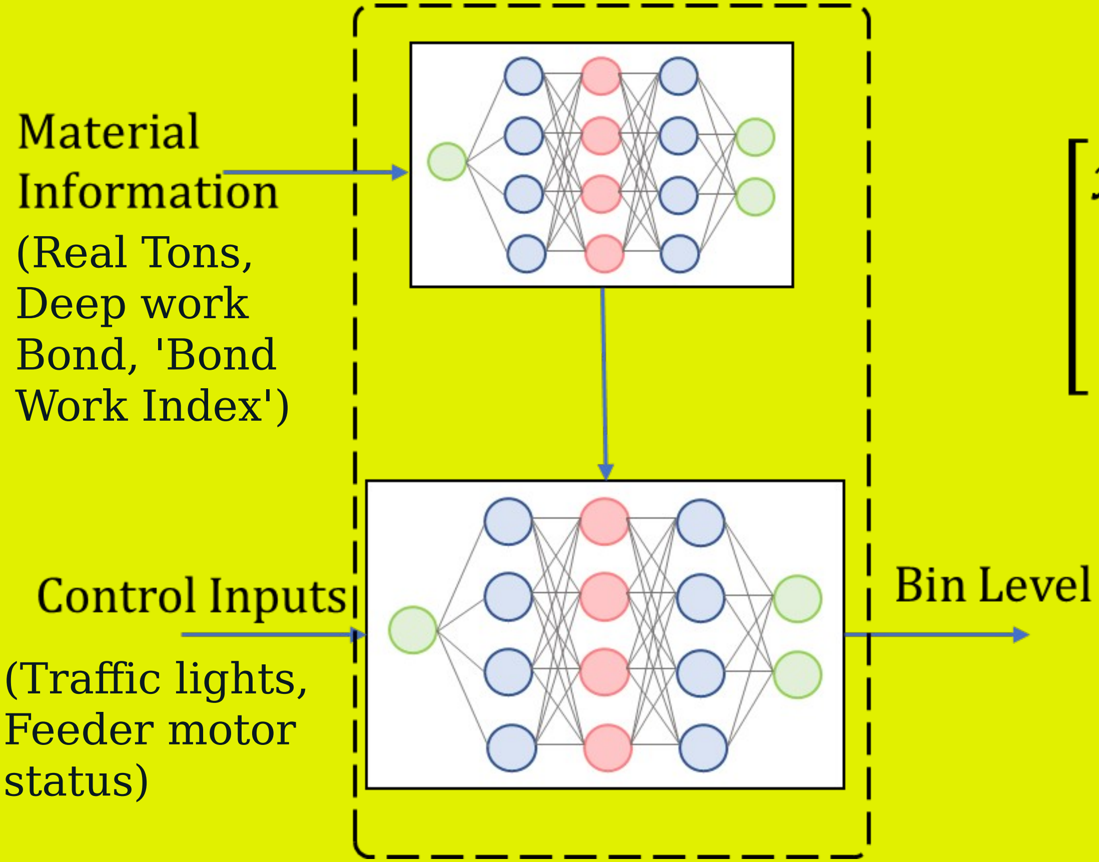
A Siemens Business

Recurrent Equilibrium Network



INNOMOTICS

Neural Network to Identify G1


Material Information
(Real Tons, Deep work Bond, 'Bond Work Index')
Control Inputs
(Traffic lights, Feeder motor status)
Bin Level

Mathematical Model of the Network

$$\begin{bmatrix} x_k(k+1) \\ w_k(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x_k(k) \\ w_k(k) \\ u(k) \end{bmatrix}$$

$$w_k(k) = \phi_k(v_k(k))$$

```
class RENLayer(nn.Module):
```

Pre-processing

INNOMOTICS

Merge of dispatch + data crusher by timestamp

Dumping time for a cycle -> green light

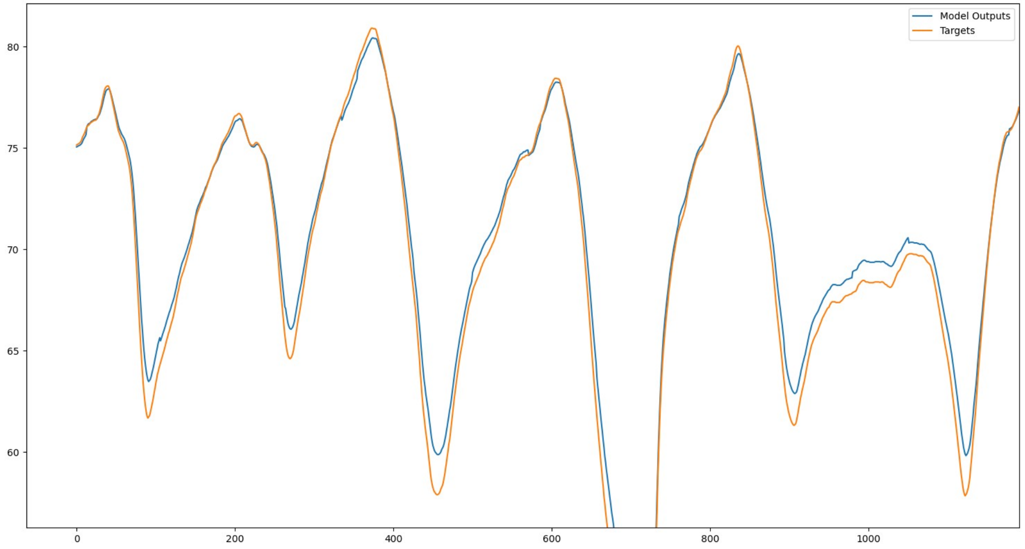
data_preprocessing_REN.py

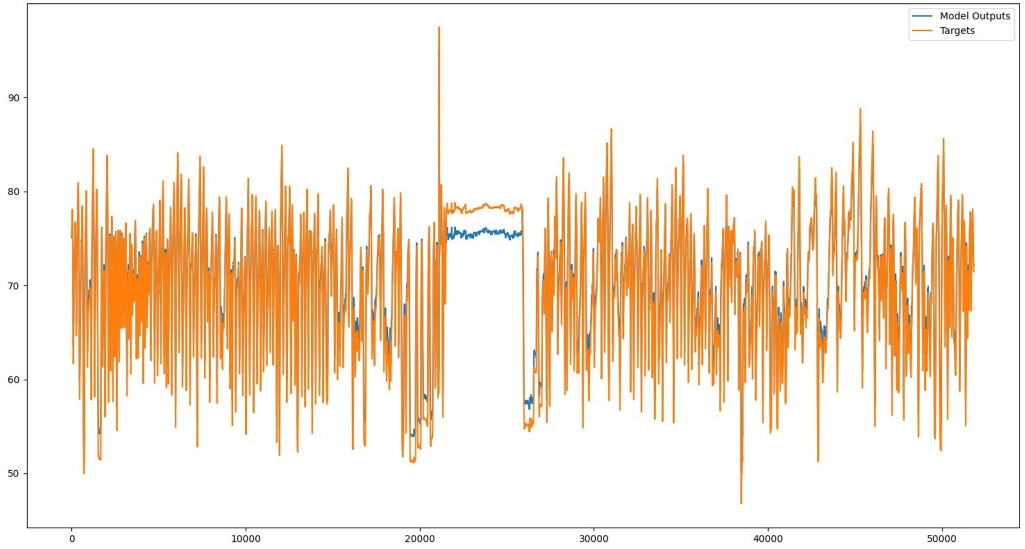
Variable	CSV Columns	Type	Purpose
first_rows	['Real Tons', 'Bond Work Index', 'Deep Work Index']	Static Inputs	One-time properties (material characteristics).
control_inputs	['traffic_light', 'feeder 1st Motor']	Time-Varying Inputs	Control signals that change over time (operational commands).
Bin_level_input	'Bin Level' (current value)	Recurrent State	Current state of the system (bin fill level).
outputs	'Bin Level' (next value)	Target	The value to predict (next bin level).

Results

INNOMOTICS

Results: Model able to predict the bin level!





Thank you.

INNOMOTICS

A Siemens Business